





# OTHER DOCUMENTS (Including Author, Title, Date, Pertinent Pages, Etc.)

AB2	Consel, C. A tour of Schism: A partial evaluation system for higher-order applicative languages. In <i>Proceedings of PEPM'93, Second ACM SIGPLAN Symposium on Partial Evaluation and Semantics-Based Program Manipulation</i> (1993), pp.145-154.
AB3	Dean, J., and Chambers, C. Optimization of object-oriented programs using static class hierarchy analysis. Tech. Rep. 94-12-01, Department of Computer Science, University of Washington at Seattle, December, 1994.
AB4	DeFouw, G., Grove, D., and Chambers, C. Fast interprocedural class analysis. In <i>Conference Record of the Twenty-Fifth ACM Symposium on Principles of Programming Languages</i> (San Diego, CA, January 1998), pp.222-236.
AB5	Emami, M., Ghiya, R., and Hendren, L.J. Context-sensitive interprocedural points-to analysis in the presence of function pointers. In <i>Proceedings of ACM SIGPLAN 1994 Conference on Programming Language Design and Implementation</i> (1994), pp. 242-256.
AB6	Fähndrich, M., and Aiken, A. Program analysis using mixed term and set constraints. In <i>Proceedings of SAS'97, International Static Analysis Symposium</i> (1997), Springer-Verlag (LNCS), pp. 114-126.
AB7	Fähndrich, M., Foster, J.S., Su, Z., and Aiken, A. Partial online cycle elimination in inclusion constraint graphs. In <i>Proceedings of ACM SIGPLAN 1998 Conference on Programming Language Design and Implementation</i> (1998), pp. 85-96.
AB8	Foster, J.F., Fähndrich, M., and Aiken, A. Polymorphic versus monomorphic flow-insensitive points-to analysis for C. In <i>Proceedings of SAS 2000, 7th Static Analysis Symposium</i> (2000), J. Palsberg, Ed., pp. 175-198.
AB9	Goldberg, A., and Robson, D. <i>Smalltalk-80 - The Language and its Implementation</i> . Addison-Wesley, 1983.
AC1	Grove, D., DeFouw, G., Dean, J., and Chambers, C. Call graph construction in object-oriented languages. In <i>Proceedings of the Twelfth Annual Conference on Object-Oriented Programming Systems, Languages, and Applications (OOPSLA '97)</i> (Atlanta, GA, 1997), pp. 108-124. <i>SIGPLAN Notices</i> 32(10).
AC2	Grove, D., DeFouw, G., Dean, J., and Chambers, C. Call graph construction in object-oriented languages. In <i>Proceedings of OOPSLA '97, ACM SIGPLAN Conference on Object-Oriented Programming Systems, Languages, and Applications</i> (1997), pp. 108-124. <i>SIGPLAN Notices</i> 32(10).
AC3	Heintze, N. Set-based analysis of ML programs. In <i>Proceedings of ACM Conference on LISP and Functional Programming</i> (1994), pp. 306-317.
AC4	Henglein, F. Dynamic typing. In <i>Proceedings of ESOP'92, European Symposium on Programming</i> (1992), Springer-Verlag (LNCS 582), pp. 233-253.
AC5	Ishizaki, K., Kawahito, M., Yasue, T., Komatsu, H., and Nakatani, T. A study of devirtualization techniques for a Java just-in-time compiler. In <i>Proceedings of the Fifteenth Annual Conference on Object-Oriented Programming Systems, Languages, and Applications (OOPSLA'00)</i> (Minneapolis, Minnesota, 2000).
AC6	Ishizaki, K., Kawahito, M., Yasue, T., Takeuchi, M., Ogasawara, T., Suganuma, T., Onodera, T., Komatsu, H., Nakatani, T. Design, implementation, and evaluation of optimizations in a just-in-time compiler. In <i>Proceedings of the ACM SIGPLAN JavaGrande Conference</i> (San Francisco, CA, June 1999).
AC7	Jagannathan, S., and Weeks, S. A unified treatment of flow analysis in higher-order languages. In <i>Proceedings of POPL'95, 22nd Annual SIGPLAN-SIGACT Symposium on Principles of Programming Languages</i> (1995), pp. 393-407.
AC8	Jagannathan, S., and Wright, A. Effective flow analysis for avoiding run-time checks. In <i>Proceedings of SAS'95, International Static Analysis Symposium</i> (Glasgow, Scotland, September 1995), Springer-Verlag (LNCS 983).
AC9	Jones, N., and Muchnick, S. A flexible approach to interprocedural data flow analysis of programs with recursive data structures. In <i>Ninth Symposium on Principles of Programming Languages</i> (1982), pp. 66-74.
AD1	Nielson, F., and Nielson, H.R. Infinitary control flow analysis: A collecting semantics for closure analysis. In <i>Proceedings of POPL'97, 24th Annual SIGPLAN-SIGACT Symposium on Principles of Programming Languages</i> (1997), pp. 332-345.
AD2	Oxhøj, N., Palsberg, J., and Schwartzbach, M.I. Making type inference practical. In <i>Proceedings of ECOOP'92, Sixth European Conference on Object-Oriented Programming</i> (Utrecht, The Netherlands, July 1992), Springer-Verlag (LNCS 615), pp. 329-349.
AD3	Palsberg, J., and Schwartzbach, M.I. Object-oriented type inference. In <i>Proceedings of OOPSLA'91, ACM SIGPLAN Sixth Annual Conference on Object-Oriented Programming Systems, Languages and Applications</i> (Phoenix, Arizona, October 1991), pp. 146-161.

OTHER DOCUMENTS (Including Author, Title, Date, Pertinent Pages, Etc.)

AD4	Schmidt, D. Natural semantics-based abstract interpretation. In <i>Proceedings of SAS'95, International Static Analysis Symposium</i> . (Glasgow, Scotland, September 1995), Springer-Verlag (LNCS 983).
AD5	Shapiro, M., and Horwitz, S. Fast and accurate flow-insensitive points-to analysis. In <i>Conference Record of the Twenty-Fourth ACM Symposium on Principles of Programming Languages</i> (Paris, France, 1997), pp. 1-14.
AD6	Sharir, M., and Pnueli, A. Two approaches to interprocedural data flow analysis. In <i>Program Flow Analysis, Theory and Applications</i> , S. Muchnick and N. Jones, Eds. 1981.
AD7	Srivastava, A. Unreachable procedures in object oriented programming. <i>ACM Letters on Programming Languages and Systems</i> , 1, 4 (December 1992), pp. 355-364.
AD8	Steensgaard, B. Points-to analysis in almost linear time. In <i>Proceedings of the Twenty-Third ACM Symposium on Principles of Programming Languages</i> (St. Petersburg, FL, January, 1996), pp. 32-41.
AD9	Stefanescu, D., and Zhou, Y. An equational framework for flow analysis of higher-order functional programs. In <i>Proceedings of ACM Conference on LISP and Functional Programming</i> (1994), pp. 318-327.
AE1	Su, Z., Fahndrich, M., and Aiken, A. Projection merging: Reducing redundancies in inclusion constraint graphs. In <i>Proceedings of POPL'00, 27th Annual SIGPLAN-SIGACT Symposium on Principles of Programming Languages</i> (2000), pp. 81-95.
AE2	Sundaresan, V., Hendren, J., Razafimahefa, C., Vallée-Rai, R., Lam, P., Gagnon, E., and Godin, C. Practical virtual method call resolution for Java. In <i>Proceedings of the Fifteenth Annual Conference on Object-Oriented Programming Systems, Languages, and Applications (OOPSLA'00)</i> (Minneapolis, Minnesota), 2000.
AE3	Sweeney, P.F., and Tip, F. Extracting library-based object-oriented applications. In <i>Proceedings of the Eighth International Symposium on the Foundations of Software Engineering (FSE-8)</i> (November 2000). To appear. A previous version of this paper appeared as IBM Research Report RC 21596, November 1999.
AE4	Tip, F., Laffra, C., Sweeney, P.F., and Streeter, D. Practical experience with an application extractor for Java. In <i>Proceedings of the Fourteenth Annual Conference on Object-Oriented Programming Systems, Languages, and Applications (OOPSLA'99)</i> (Denver, CO), 1999, pp. 292-305, <i>SIGPLAN Notices</i> 34(10).
AE5	Vallée-Rai, R., Gagnon, E., Hendren, L., Lam, P., Pominville, P., and Sundaresan, V. Optimizing java bytecode using the soot framework: Is it feasible? In <i>Proceedings of CC'00, International Conference on Compiler Construction</i> (2000), Springer-Verlag (LNCS).
AE6	Vitek, J., Horspool, R.N., and Krall, A. Efficient type inclusion tests. In <i>Proceedings of OOPSLA '97, ACM Systems, Languages and Applications</i> (1997), pp. 142-157. <i>SIGPLAN Notices</i> 32(10).

Examiner:

UYEN LE

Date Considered:

20 MAR 04

EXAMINER: Initial if reference considered, whether or not citation is in conformance with MPEP 609; Draw line through citation if not in conformance and not considered. Include copy of this form with next communication to applicant.

RECEIVED

JUN 7 2001

Technology Center 2100